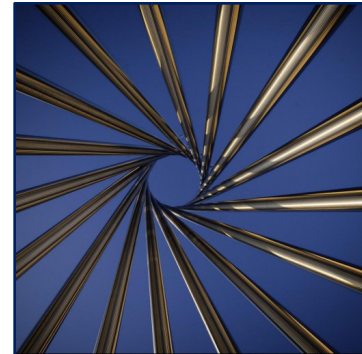




Status of *art* 3.01

9 August 2018
art stakeholders meeting



General status

- *art* 3.01 will be C++17 only
- ROOT support is now provided through `art_root_io`
- We intend to discontinue c2 in favor of a newer Clang version (c5 or c7)
- Some hiccups with moving to new versions of Clang
 - We are working through them
- We are waiting on a resolution to a ROOT 6.14/04 bug before we release

-e|--estart program option

- The -e <#> option has been provided to allow processing from event <#> *in Run 1, SubRun 0*.
- This has caused frustration over the years:
 - *“I want to process event 7 from Run 12, SubRun 8, in this file.”*
- For *art* 3.01, the proposal is to deprecate the single-number specification, and support a triplet specification:

```
[knoepfel@woof]$ art -c /dev/null -n 10 -e 6
...
Specifying an event number of 6 is now deprecated when using
the -e|--estart program option. Please explicitly specify the
run and subrun numbers (e.g.):
  -e "1:0:6"
...
```

Collisions of preprocessor macros

- Preprocessor macros are **not** encapsulated by namespaces

```
namespace genie {  
#define LOG_DEBUG ...  
}  
namespace mf {  
#define LOG_DEBUG ... // ! Collision  
}
```

- Convention is to preface the relevant macros with a package prefix:

```
namespace genie {  
#define GENIE_LOG_DEBUG ...  
}  
namespace mf {  
#define MF_LOG_DEBUG ... // Okay  
}
```

Collisions of preprocessor macros

- Preprocessor macros are **not** encapsulated by namespaces

```
namespace genie {  
#define LOG_DEBUG ...  
}  
  
namespace mf {  
#define LOG_DEBUG // ! Collision
```

- Conversion

Proposal is that beginning with *art* 3.01, all user-facing messagefacility macros will have an **MF_** prefix.

- Migration script will be provided

An *art* 2.12 release will also be provided that incorporates only this breaking change.

```
#define MF_LOG_DEBUG ... // Okay  
}
```

Deprecating ART_DEBUG_CONFIG environment variable

- The ART_DEBUG_CONFIG environment variable has largely been superseded by
`art --debug-config=out.fcl ...`
- One difference is that ART_DEBUG_CONFIG supports writing to STDOUT via
`ART_DEBUG_CONFIG=1 art ...`
- To clean things up, the proposal is that for *art* 3.01:
 - Support `--debug-config=STDOUT`, which is equivalent to `ART_DEBUG_CONFIG=1`
 - Deprecate the ART_DEBUG_CONFIG environment variable